TDB-ACC-NO: NN9603363

DISCLOSURE TITLE: **Anonymous Delivery of Goods in Electronic Commerce**

PUBLICATION-DATA: IBM Technical Disclosure Bulletin, March 1996, US

VOLUME NUMBER: 39

ISSUE NUMBER: 3

PAGE NUMBER: 363 - 366

PUBLICATION-DATE: March 1, 1996 (19960301)

CROSS REFERENCE: 0018-8689-39-3-363

DISCLOSURE TEXT:

Disclosed is a method that allows on-line (electronic) purchase and delivery of (both electronic and physical) goods in a manner that preserves anonymity of the consumer. The method is secure and resistant to cheating by both consumers and merchants.

The following notation is used throughout this document.

C,M - Consumer and Merchant, the protocol participants;
ID-x - user ID of X;
PK-x - Public Key of X (X=C or X=M);
SK-x - Secret/Private Key of X (X=C or X=M);
Rc/Rd - Random numbers (nonces)
Cert-x - Public Key Certificate of X; includes PK-x
H(text) - Strong one-way Hash function computed over "text", e.g., Secure Hash Function (SHA) or MD5.

Sx(text)- Signature computed under SKx, Sx text =SK-x(H(text))
[text] - Optional text

Prerequisites for the present method are the possibility of anonymous communication (e.g., (1) and a public key infrastructure (for merchants only). The buying process is started by a sender, usually a prospective consumer, who composes an offer request with a plaintext (unencrypted) description of the desired product or service and a random quantity H(Rc). This construction does not by itself reveal the sender's identity.

The resultant offer request is sent anonymously to one or more selected merchant(s), or even broadcasted, via the network. If a merchant decides to make an offer, he/she composes a reply with an offer description and his/her digital signature (SIG_offer), which is computed over the sender's random quantity H(Rc), and transmits it back to the sender. The merchant's public key may also have to be transmitted since in some cases the sender does not yet have it.

Upon receiving the message, the consumer can (if necessary) extract the merchant's public key and verify the merchant's SIG_offer computed over (among other values) the consumer's H(Rc).

The present method commences when the consumer decides to purchase the aforementioned merchandise based on a previous bid/offer. The payment process itself is outside the scope of this document.

(See (2) or (3) for examples of secure electronic payment

protocols and scenarios.)

Assume that the payment process takes place before the delivery of goods (although it can, in principle, take place concurrently.)

Step 1.

a) (note: Consumer is assumed to retain Rc and SIG_offer from above.) Consumer generates another random number Rd and computes H(Rd).

b) Optionally, consumer generates a public/private key-pair (PKtmp,SKtmp).

(This key-pair is to be used for the delivery of goods later.)

c) Consumer sends to merchant a COMMIT_REQUEST message containing:
H(Rc), H(Rd), C_options

where H(Rc), H(Rd) are as described above and "C_options" are optional parameters including, for example: date/time-stamp, PKtmp, mailing address (for off-line, non-electronic goods), etc.

Step 2.

a) Merchant receives the COMMIT_REQUEST and extracts both H(Rc) and H(Rd). H(Rd) is stored for future reference.

b) Using H(Rc) merchant searches his records and checks if the corresponding bid/offer has been processed.

If the offer is no longer valid, merchant sends and error message to consumer and terminates processing.

c) If the offer is still valid, merchant composes and send to consumer a COMMIT message containing:

SKm COMMIT_REQUEST, M_options , M_options
_____ _____/
SIG_commit

where SIG_commit represents a merchant's signature computed with SKm over the specified data. "M_options" are optional parameters.

Step 3.

a) Consumer receives the OFFER message and (using merchant's public key PKm) verifies SIG_commit. If the signature is invalid, an error message to that effect is sent to merchant.

b) If signature is valid, consumer generates and sends to merchant a DELIVERY_REQUEST message containing: Rc

Step 4.

a) Merchant receives DELIVERY_REQUEST and extracts Rc.

b) Merchant computes TMP=H(Rc) and compares to H(Rc) stored in the appropriate transaction record. If there is no match, an error message to that effect is sent to consumer.

c) If TMP matches H(Rc) merchant composes and sends to consumer, a DELIVERY message containing:

GOODS, SKm SIG_commit,GOODS
___ ____/
SIG_deliver

or, if PKtmp was included in COMMIT_REQUEST message (as in Step 1 above), a DELIVERY message contains:

PKtmp(GOODS), SKm SIG_COMMIT,GOODS
___ ____/
SIG_deliver

where PKtmp(GOODS) denotes the encryption of GOODS under the consumer-generated public key PKtmp.
(Only if PKtmp was included in COMMIT_REQUEST above.)

Step 5.

a) Consumer receives DELIVERY and, if applicable, decrypts PKtmp(GOODS) using SKtmp. (Otherwise, GOODS arrives in the clear.)

b) Using PKm and SIG_commit (received in Step 3) consumer verifies SIG_deliver. If SIG_deliver is invalid an error message to that effect is sent to merchant.
Otherwise, consumer sends to merchant a TERMINATE message containing Rd.

Step 6.

a) Merchant receives TERMINATE, extracts Rd, computes TMP=H(Rd) and compares it with H(Rd) received in COMMIT_REQUEST (see step 2.) If they match the transaction is terminated. Otherwise, an error message is sent to consumer (perhaps along with the re-transmission of DELIVERY.)

The method presented above provides protection against dishonest behavior by either merchants or consumers involved. Potential cases of cheating and disputes are addressed below. All cases require intervention of a mutually trusted off-line authority that we refer to as COURT.

While dispute resolution is likely to take place off-line, it is expected that consumer will remain anonymous with respect to merchant. However, consumer may be required to reveal his identity to COURT.

At the end of a successful transaction the parties involved must have the following in their possession:

Rc, Rd, SIG_Commit, SIG_offer, H(Rc), H(Rd)

Dispute Scenarios

a) Customer is asked to produce a valid SIG_offer
   a. No valid SIG_offer; merchant prevails.
   b. Valid SIG_offer; continue with (2).

b) Merchant is asked to provide Rc.
   a. Merchant can not produce the correct Rc; continue with (3).
   b. Correct Rc; continue with (4)

c) Consumer is asked to produce Rc.
   a. Correct Rc; consumer prevails (protocol can be re-run.)
   b. Incorrect or no Rc; merchant prevails.

d) Consumer is asked to produce Rc.
   a. Correct Rc; continue with (5).
   b. Incorrect or no Rc; merchant prevails.

e) Consumer is asked to produce a valid SIG_commit.
   a. No valid SIG_commit; merchant prevails.
   b. Valid SIG_commit; continue with (6).

f) Merchant is asked to produce Rd.
   a. Correct Rd; merchant prevails.

   b. Incorrect or no Rd; consumer prevails (merchant is ordered

to send a DELIVERY message to consumer and consumer is ordered
to reply with a TERMINATE message (the latter containing a valid Rd.)

References

(1) C. Gulcu and G. Tsudik, "Mixing E-mail with BABEL," 1996 Symposium on Network and Distributed System Security (February 1996).

(2) M. Bellare, R. Hauser, A. Herzberg, J. Garay, H. Krawczyk, M. Steiner, G. Tsudik and M. Waidner, "iKP -- A Family of Secure Electronic Payment Protocols," USENIX Conference on Electronic Commerce (July 1995).

(3) D. Chaum, A. Fiat and M. Naor, "Untraceable Electronic Cash," In Proceedings of Crypto'88, Santa Barbara, Ca. (August 1988).